REMARKS/ARGUMENTS

Claims 1 - 34 are pending. Claims 1 - 34 are rejected. Claims 1, 3, 4, 5, 10, 20, 22, 24-26, 31-32 are amended herein.

As discussed below, all of the claims are in condition for allowance. **But if after considering this response, the Examiner does not allow all of the claims, then the Applicant's agent formally requests that the Examiner contact him to schedule and conduct a telephone interview before issuing a subsequent office action.**

Rejection of Claims 5 and 24 Under 35 U.S.C. § 112 ¶2

Claim 5

On page 2 of the Office Action, the Examiner asserts that the language of claim 5 wherein "the single action is a double-click," is unclear. Applicant's agent traverses the rejection and asserts that one of ordinary skill in the art would recognize that the double-clicking of a mouse or other pointing device is a single action in a modern computing environment. It is well known that a double-click with a computer pointing device is a separate type of interaction than, for instance, a "single click" or a "right-click."

Applicant's agent has amended the claim to clarify the nature of the limitation and to overcome the rejection, and respectfully requests that the Examiner withdraw the rejection.

Claim 24

On page 2 of the Office Action, the Examiner asserts that the language of claim 24 is unclear. Applicant's agent has amended the claim to overcome the objection, and respectfully requests that the Examiner withdraw the respective rejection accordingly.

Rejection of Claims 22-25 Under 35 U.S.C. § 101 As Being Directed to Non-Statutory Subject Matter

Claim 22

Claim 22 has been amended. Applicant's agent respectfully requests that the Examiner withdraw the rejection.

Claim 23

Claim 23 is allowable at least by virtue of its dependency from claim 22.

Claims 24-25

Claims 24-25 have been amended. Applicant's agent respectfully requests that the Examiner withdraw the rejections of claims 24-25.

Rejection of Claims 1-10, 20, 24, 26, 27, 31-33 Under 35 U.S.C. § 102(e) Over Halpern et al. (U.S Pat. No. 6,282,711)

Claim 1

Claim 1 is amended for reasons not related to allowability. Claim 1 recites receiving, from a user enabled electronic device, an input file to be used in creating a self-extracting file; and without further action by the user enabled electronic device, creating a self-extracting file using the input file. Claim 1 also recites that the input file is configured to automatically launch upon execution of the self-extracting file.

For example, referring to FIGS. 2 and 4 and page 16, line 20, through page 20, line 24, of the patent application, an input file 202 (FIG. 2) is received in step 402 from a user-enabled electronic device. In steps 404-420, a self-extracting file 204 (FIG. 2) is created without any further action by the user-enabled electronic device. Referring further to FIG. 6 and pages 20-21 of the application, input file 202 is automatically launched upon execution of the output self-extracting file 204.

In contrast, Halpern does not disclose receiving, from a user enabled electronic device, an input file to be used in creating a self-extracting file. Halperan also does not disclose creating a self-extracting file using the input file without further action by the user-enabled electronic device.

Halpern discloses a system and method for packaging and distributing software by first allowing a user to select a subset of software application “components and

options" available for installation, and then creating on a remote server a self-extracting installation package for distributing to the user files that provide the selected subset of the software application. Referring, e.g., to FIGS. 2-3, col. 3 lines 39-54, and col. 5 lines 49-55, Halpern does not disclose receiving an input file from a user-enabled electronic device to be used in creating a self-extracting file.

Contrary to the Examiner's contention Halpern column 5, lines 41-44, do not disclose receiving an input file from a user-enabled electronic device. Rather, column 5, lines 41-44 of Halpern discloses providing one of three selected user interfaces, UI-1, UI-2, or UI-3 to the user for selecting components and options of the software application. Nowhere does Halpern disclose that an input file is received from a user-enabled electronic device.

Referring, e.g., to column 3 line 62 through column 4 line 18, Halpern discloses that once a user selects the components and options relevant to the desired software program, an executable file is created and transmitted to the user such that the user may locally install the desired software program by executing the executable file.

Selection of components and options is apparently a plurality of actions. Following Halpern's selection of components and options, an options manager retrieves metadata from a database, an installer set generator accesses a component pool to produce a custom set of files corresponding to the user's selections, an installer set generator submits the custom set of files to a packager, and the packager appends a client installer and submits the combined custom files and installer to a compressor [column 7, lines 23 – 38].

Finally, claim 1 recites that the input file is automatically launched upon execution of the self-extracting file.

In contrast, Halpern indicates that the user executes the retrieved file *and* runs the client installer [column 7, lines 55-56]. By comparing Halpern's Step 7 to Step 12, it is apparent that while Halpern's transmitted compressed file is self-extracting, the client installer program is a separate entity. Thus, Halpern's input file is not disclosed to be automatically launched upon execution of the self-extracting file. Rather, running the client installer is apparently a separate process.

Thus, the file for compression is not received from a user enabled device, but

rather is retrieved from a component pool. Creation of a self-extracting file is not disclosed to be performed without further action by the user enabled device, because Halpern requires selection of a plurality of "components and options" within the user interfaces UI-1, UI-2, or UI-3. And the self-extracting file is not disclosed to launch the input file automatically upon execution of the self-extraction.

Accordingly, Halpern does not disclose all the limitations of claim 1, and claim 1 is allowable over Halpern.

Claim 2

Claim 2 recites automatically generating a filename for the self-extracting based in part on the filename of the received input file.

Claim 2 is allowable by virtue of its dependency from claim 1.

Moreover, Halpern does not disclose naming a self-extracting file based in part on the filename of a received input file (even if, *arguendo*, the input file was received from the user enabled electronic device). Halpern is apparently silent on the names of input files, and also does not disclose naming a self-extracting file based in part on the name of an input file.

Claim 2 is also allowable over Halpern for at least this additional reason.

Claim 3

Claim 3 is amended for reasons not related to allowability. Claim 3 recites receiving an input file and creating a self-extracting file in response to only a single action from a user.

In contrast, Halpern discloses receiving an indication of program components and options to be installed on a user computer from one of several user interface templates UI-1, UI2, and UI-3 [Column 5, lines 41-44]. However, Halpern does not disclose that any of the interface templates UI-1, UI-2, or UI-3 include selection of an input file for compression. Rather, the user selects "program components and options". Typically, modern programs include a large number of program components. "Program components and options" are typically not, and are not disclosed by Halpern, to correspond to "an input file." Rather, Halpern's options manager 104 retrieves meta information from a component information database to relate the selected components and options to a custom set of installation files. [Column 7, lines 22-26]

Moreover, Halpern does not disclose selection of program components and options to be only a single action. Indeed, Halpern refers to program components and options in plural form throughout, and discloses user selection to be selection of desired software components and options from a list of component and options [Column 7, lines 23-24].

Finally, as described in conjunction with claim 1, Halpern also does not disclose an input file being automatically launched upon execution of a self-extracting file.

Accordingly, Halpern does not disclose all the limitations of claim 3, and claim 3 is allowable over Halpern.

Claims 4-7

Claims 4 and 5 are amended to overcome section 112 rejections. Claims 4-7 each recite a type of single action that initiates creation of a self-extracting file.

Claims 4-7 are allowable by virtue of their dependence from claim 3.

Moreover, since Halpern does not disclose creating a self-extracting file from only a single action, it also does not disclose the types of single actions by a user recited by claims 4-7, and claims 4-7 are also allowable for this additional reason.

Claim 8

Claim 8 is allowable by virtue of its dependence from claim 3.

Claim 9

Claim 9 is allowable by virtue of its dependence from claim 3.

Claim 9 recites generating a filename for the self-extracting file based on the filename of an input file. Claim 9 is additionally allowable for reasons similar to those given for claim 2. Namely, Halpern is silent with respect to how the name of the self-extracting file may be based or not be based on the name of an input file.

Claim 10

Claim 10 is amended.

Claim 20

Claim 20 recites creating a self-extracting file from an input file, wherein the input file is one of a plurality of file types, and automatically selecting a loader based on the input file's type, wherein the input file will be automatically launched upon execution of the self-extracting file.

As discussed above with respect to claims 1 and 3, Halpern does not disclose an input file being automatically launched upon execution of a self-extracting file.

Moreover, Halpern does not disclose automatically selecting a loader based on the input file's type. On page 7 of the Office Action, the Examiner asserts that "the limitations of claims 10, 20, 26 and 32 are rejected in the analysis of Claim 1 above, and these claims are rejected on that basis." However, claim 1 does not recite automatically selecting a loader based on an input file's type. If the Examiner maintains the rejection against claim 20, Applicant's agent requests that he specify which portions of Halpern anticipate the recited limitations of the claim.

Claim 24

Claim 24 recites a system for receiving a user a selection of an input file to be used in creating a self-extracting file, compressing the received input file, and creating, in response to only a single action by a user, an executable file from the compressed input file wherein the input file is automatically launched upon execution of the executable file.

As discussed above with respect to claims 1 and 3, Halpern does not disclose receiving from a user a selection of an input file. Halpern only discloses allowing a user to select a subset of components and options related to a desired software program, after which a subset of files relevant to those components and options are packaged and distributed to the user.

Moreover, for reasons similar to those given for claim 1, Halpern does not disclose an executable file, wherein the input file is automatically launched upon execution of the executable file. Accordingly, Halpern does not disclose all the limitations of claim 24, and claim 24 is allowable over Halpern.

Claim 26

Claim 26 recites receiving an input file from a user to be used in creating a self-extracting file, and automatically creating a self-extracting file.

In contrast, and as discussed above with respect to claim 1, Halpern does not disclose receiving from a user an input file to be used in creating a self-extracting file. Halpern discloses that a user is allowed to select a subset of program “components and

options" available for installation, after which a subset of files corresponding to those components and options are packaged and distributed to the user. Halpern does not disclose receiving any file from a user. Accordingly, Halpern does not disclose all the limitations of claim 26, and claim 26 is allowable over Halpern.

Claim 27

Claim 27 is allowable at least for reasons similar to those given for claims 1 and 3. Namely, Halpern does not disclose an input file being automatically decompressed and launched upon execution of an executable file. Halpern also does not disclose receiving an input file in response to a single action. Accordingly, Halpern does not disclose all the limitations of claim 27, and claim 27 is allowable over Halpern.

Claim 31

Claim 31 is amended. Claim 31 recites executing an executable file to decompress a compressed copy of an input file, and launching the decompressed input file with appropriate application software.

In contrast, as discussed above with respect to claims 1 and 3, Halpern does not disclose launching a decompressed input file with appropriate application software, but rather discloses decompressing a desired software program by executing an executable file. Halpern does not disclose that the software program is launched with appropriate application software after the executable file is executed. Accordingly, Halpern does not disclose all the limitations of claim 31, and claim 31 is allowable over Halpern.

Claim 32

Claim 32 is amended. Claim 32 is allowable at least for reasons discussed above with respect to claims 1 and 3.

Claim 33

Claim 33 is allowable at least by virtue of its dependency from claim 32.

Rejection of Claims 21-23 Under 35 U.S.C. § 102(e) Over Wygodny et al. (U.S Pat. No. 6,202,199)

Claim 21

Claim 21 recites displaying a first frame used to allow a user to specify an input file to be converted to a self-extracting file, receiving the input file specified by the user,

wherein the received input file is automatically configured as a self-extracting file, and wherein the input file is automatically launched upon execution of the self-extracting file, and displaying a second frame, wherein the second frame includes a link related to the self-extracting file created from the user specified input file.

In contrast, Wygodny does not disclose displaying a first frame used to allow a user to specify an input file to be converted to a self-extracting file, does not disclose automatically configuring an input file specified by the user as a self-extracting file, and does not disclose displaying a second frame that includes a link related to a self-extracting file created from the user-specified input file. Wygodny discloses a system for tracing the execution paths of a software program without requiring modifications to the executable or source code files of that program. As part of this system, Wygodny teaches providing a small executable "agent" program that enables a remote user to generate a trace file, and further teaches that the preferred method of providing this "agent" program includes packaging that program as a self-extracting file. These are not the limitations of claim 21.

On page 11 of the instant Office Action, the Examiner cites Wygodny's FIG. 3A and “col 8, line 51-55, col 17, line 1-7, fig 3A, fig 9-10” to show displaying a first frame used to allow a user to specify an input file to be converted to a self-extracting file. Respectfully, this is incorrect. Referring, *e.g.*, to FIG. 3A, column 8 lines 50-57 and column 9 lines 9-13, Wygodny's frame window 300 (and specifically, executable pane 314) depicts a particular executable file that is currently being traced. It does not allow a user to specify an input file to be converted to a self-extracting file.

Also on page 11 of the instant Office Action, the Examiner cites column 17 lines 1-12 to show that Wygodny discloses automatically configuring an input file specified by the user as a self-extracting file. Respectfully, this is also incorrect. Referring to column 17 lines 1-12, Wygodny teaches supplying a tracing "agent" program to a user as a self-extracting "zip file.” To install this "agent" program, the user can simply double-click on that zip file. Finally, the user may run the "agent" program, which allows the user to specify both a "Trace Control Information" (TCI) file and a client executable program which the user desires to have traced. These are not the limitations of claim 21. Wygodny does not disclose automatically configuring an input file specified by the

user as a self-extracting file.

The Examiner further cites FIGS. 3A-3B and FIG. 5 to show that Wygodny teaches displaying a second frame that includes a link related to a self-extracting file created from a user-specified input file. Respectfully, this is incorrect. Referring to FIGS. 3B and 5, although Wygodny does teach displaying a second frame (relative to the first frame depicted in FIG. 3A) as part of its user interface, none of the user interface frames depicted include a link related to a self-extracting file that has been created from a user-specified input file.

Wygodny does not disclose all the limitations of Claim 21, and Claim 21 is therefore allowable over Wygodny.

Claim 22

Claim 22 is amended. Claim 22 recites a receiving module configured to receive an input file, wherein the input file received is one of a plurality of file types, and a naming module configured to create and name an output file.

In contrast, Wygodny does not disclose receiving an input file, wherein the input file received is one of a plurality of file types. On page 12 of the instant Office Action, the Examiner cites “col 9, line 9-13, line 57-62, col 12, line 24-35” to show that Wygodny teaches this limitation. However, to the extent that Wygodny discloses any "receiving module," it does so only with respect to the ability to receive executable files in order to trace the operations of those executable files. Referring, *e.g.*, to column 9 lines 9-62 and column 12 lines 24-35, the executable files selectable as input files have a single file format.

Contrary to the Examiner’s assertion, Wygodny does not disclose a naming module wherein an output filename is generated from the associated filename of the input file. Claim 22 is also allowable for this additional reason.

Claim 22 further recites a self-extracting module configured to transform an output file into a executable file. Although the Examiner does not provide a citation to show this limitation, Applicant's agent can find no instance within the Wygodny reference that teaches transforming an output file into an executable file. Referring to column 16 lines 41-44 and column 17 lines 1-12, Wygodny discloses distributing the tracing agent program to the user as a self-extracting zip file. That tracing agent program, however, is not "an output file" as recited by claim 22.

Wygodny does not disclose all the limitations of claim 22, and Claim 22 is allowable over Wygodny.

Claim 23

Claim 23 is allowable at least by virtue of its dependency from claim 22.

Rejection of Claims 11-19, 25, 28-29, 30, 34 Under 35 U.S.C. § 103(a) Over Halpern et al. And Further In View Of Gage et al. (U.S. Pat. No. 5,923,846)

Claims 11-19

Claims 11-19 are allowable at least by virtue of their dependencies from claim 10.

Claim 25

Claim 25 as amended recites a compressed input data portion corresponding to an input data file, and a self-extracting stub portion that includes a loader operable to launch the decompressed input data portion with appropriate application software for handling the input data file.

For example, referring to FIG. 2 and corresponding text of the patent application, in an embodiment a compressed input data portion 206 corresponds to an input data file 202, and a self-extracting stub portion 210 that includes a loader operable to launch the decompressed input data portion with appropriate application software for handling the input data file.

In contrast, and as discussed above with respect to the rejections of claims 1 and 3, Halpern neither discloses nor renders obvious a loader operable to launch a decompressed input data portion with appropriate application software for handling an input data file.

Gage does not provide the missing disclosure. Gage discloses a system for posting and downloading textual messages and files from an online bulletin board system. Referring, *e.g.*, to FIG. 5 and corresponding text, Gage teaches storing textual messages as compressed RTF (Rich text format) data having a plurality of compressed portions and an uncompressed header portion. However, Gage does not disclose or suggest utilizing self-extracting files as part of its online bulletin board system, and is therefore silent as to the use of including a loader operable to launch a decompressed input data portion with appropriate application software.

Neither Halpern nor Gage, either alone or in combination, disclose or reasonably suggest the limitations of claim 25, and Claim 25 is allowable over Halpern and Gage.

Claims 28-30

Claims 28-30 are allowable for reasons similar to those given for claim 25.

Claim 34

Claim 34 is allowable at least by virtue of its dependency from claim 32.

In the event additional fees are due as a result of this amendment, the Commissioner is hereby authorized to charge any deficiency of fees submitted herewith, or credit any overpayment, to Deposit Account No. 07-1897.

If the Examiner believes that a telephone interview would be helpful, he is respectfully requested to contact the Applicants' agent at (425) 455-5575.

Dated this 24th day of March, 2009.

Respectfully submitted,

/CAWiklof/

Christopher A. Wiklof
Registration No. 43,990

Customer No. 72455

Graybeal Jackson LLP
155 - 108th Avenue NE, Suite 350
Bellevue, Washington 98004-5973
Telephone: 425.455.5575
Facsimile: 425.455.1046